

In-network reconstruction of jointly sparse signals with ADMM

Original

In-network reconstruction of jointly sparse signals with ADMM / Matamoros, J.; Fosson, Sophie; Magli, Enrico; Antón Haro, C.. - (2015). (Intervento presentato al convegno European Conference on Networks and Communications (EUCNC 2015) tenutosi a Paris (France) nel jun 29 - jul 02, 0215).

Availability:

This version is available at: 11583/2624989 since: 2015-12-05T14:56:50Z

Publisher:

Published

DOI:

Terms of use:

openAccess

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

In-network reconstruction of jointly sparse signals with ADMM

Javier Matamoros, Sophie M. Fosson,
Enrico Magli and Carles Antón-Haro

Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)
Politecnico di Torino



EUCNC 2015, June 29- July 2, Paris (France)

Motivation

- ▶ Reconstruction of jointly sparse signals with innovations
- ▶ Sensing applications where information is acquired by a geographically distributed set of nodes
- ▶ No need for a fusion center → In-network processing
- ▶ Focus on efficient ADMM techniques with low signalling overhead

Outline

Signal Model

Centralized ADMM

Distributed ADMM

Distributed ADMM with 1 bit

Numerical results

Conclusions

Signal Model

- ▶ Consider N sensor nodes
- ▶ Observed signal at the i th sensor node

$$y_i = A_i x_i + \eta_i \quad ; \quad i \in \mathcal{N}$$

- ▶ $x_i \in \mathbb{R}^n$: Signal of interest
- ▶ $A_i \in \mathbb{R}^{M \times L}$ with $M \ll L$: Measurement matrix
- ▶ $\eta_i \in \mathbb{R}^L$: Acquisition noise

Signal Model

- ▶ Consider N sensor nodes
- ▶ Observed signal at the i th sensor node

$$y_i = A_i x_i + \eta_i \quad ; \quad i \in \mathcal{N}$$

- ▶ $x_i \in \mathbb{R}^n$: Signal of interest
- ▶ $A_i \in \mathbb{R}^{M \times L}$ with $M \ll L$: Measurement matrix
- ▶ $\eta_i \in \mathbb{R}^L$: Acquisition noise
- ▶ **Assumption:** JSM-1 model [Duarte06]

$$x_i = \Psi z_c + \Omega_i z_i \quad ; \quad i \in \mathcal{N}$$

- ▶ $z_c \in \mathbb{R}^n$: Common signal with k_c non-zero components
- ▶ $z_i \in \mathbb{R}^n$: Innovation with k_i non-zero components
- ▶ $\Psi, \Omega_i \in \mathbb{R}^{L \times L}$: Sparsity basis (w.l.g. $\Psi = \Omega_i = I_L$)

Centralized ADMM

Optimization problem

- Lasso formulation...

$$\begin{aligned} \min_{\{x_i, z_i\}, z_c} \quad & \frac{1}{2} \sum_{i=1}^N \left(\|y_i - A_i x_i\|_2^2 + \tau_1 \|z_i\|_1 + \tau_2 \|z_c\|_1 \right) \\ \text{s.t.} \quad & x_i = z_c + z_i; \quad i = 1, \dots, N \end{aligned}$$

- Promotes sparsity in the innovation component
- Promotes sparsity in the common component

Centralized ADMM

Review

► Augmented cost function

$$\begin{aligned} \min_{\{x_i, z_i\}, z_c} \quad & \frac{1}{2} \sum_{i=1}^N \left(\|y_i - A_i x_i\|_2^2 + \tau_1 \|z_i\|_1 + \tau_2 \|z_c\|_1 \right. \\ & \left. + \frac{\rho}{2} \|x_i - z_i - z_c\|_2^2 \right) \\ \text{s.t.} \quad & x_i = z_i + z_c; \quad i = 1, \dots, N \end{aligned}$$

Centralized ADMM

- ▶ Augmented cost function

$$\text{s.t.} \quad x_i = z_i + z_c; \quad i = 1, \dots, N$$

$$\begin{aligned} \mathcal{L} := & \frac{1}{2} \sum_{i=1}^N \|y_i - A_i x_i\|_2^2 + \sum_{i=1}^N \tau_1 \|z_i\|_1 + N \tau_2 \|z_c\|_1 \\ & + \sum_{i=1}^N \frac{\rho}{2} \|x_i - z_i - z_c\|_2^2 + \sum_{i=1}^N \lambda_i^T (x_i - z_i - z_c) \end{aligned}$$

Centralized ADMM

Algorithm

- ▶ ADMM iterates

$$x_i(t+1) = (\rho I + A_i^T A_i)^{-1} (A_i^T y_i + \rho(z_i(t) + z_c(t)) - \lambda_i(t))$$

$$z_c(t+1), \{z_i(t+1)\} = \arg \min_{z_c, \{z_i\}} \mathcal{L}(t+1)$$

$$\lambda_i(t+1) = \lambda_i(t) + \rho(x_i(t+1) - z_i(t+1) - z_c(t+1))$$

Centralized ADMM

Algorithm

- ▶ ADMM iterates

$$x_i(t+1) = (\rho I + A_i^T A_i)^{-1} (A_i^T y_i + \rho(z_i(t) + z_c(t)) - \lambda_i(t))$$

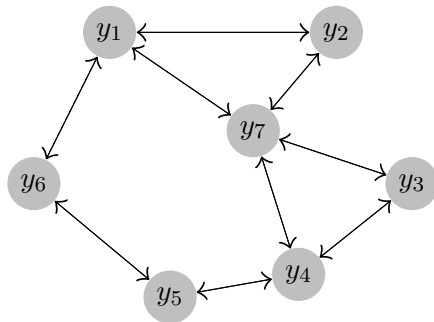
$$z_c(t+1), \{z_i(t+1)\} = \arg \min_{z_c, \{z_i\}} \mathcal{L}(t+1)$$

$$\lambda_i(t+1) = \lambda_i(t) + \rho(x_i(t+1) - z_i(t+1) - z_c(t+1))$$

- ▶ Difficult to compute (Algorithm 1 in the paper)
- ▶ **Centralized solution!**

Distributed ADMM

Distributed scenario



- ▶ Nodes only communicate with their neighbors (no fusion center)
- ▶ **Goal:** In-network reconstruction of $\{x_i\}$.

Distributed ADMM

Distributed formulation

- New formulation...

$$\begin{aligned} \min_{\{x_i, z_i, \zeta_i, c_i\}} \quad & \frac{1}{2} \sum_{i=1}^N \|y_i - A_i x_i\|_2^2 + \tau_1 \|z_i\|_1 + \tau_2 \|\zeta_i\|_1 \\ \text{s.t.} \quad & x_i = z_i + \zeta_i; \quad i \in N \\ & \zeta_i = c_j; \quad j \in \mathcal{N}_i \cup \{i\} \end{aligned}$$

- Forces consensus on the local guesses

Distributed ADMM

Distributed formulation

- Augmented cost function

$$\begin{aligned} \min_{\{x_i, z_i, \zeta_i, c_i\}} & \frac{1}{2} \sum_{i=1}^N \|y_i - A_i x_i\|_2^2 + \tau_1 \|z_i\|_1 + \tau_2 \|\zeta_i\|_1 \\ & + \frac{\rho}{2} \|x_i - z_i - \zeta_i\|_2^2 + \frac{\theta}{2} \sum_{j \in \mathcal{N}_i} \|\zeta_i - c_j\|_2^2 \\ \text{s.t.} \quad & x_i = z_i + \zeta_i; \quad i \in N \\ & \zeta_i = c_j; \quad j \in \mathcal{N}_i \cup \{i\} \end{aligned}$$

- Difficult to solve by means of classical ADMM (i.e. with 2 primal iteration blocks)

Distributed ADMM

Distributed formulation

- Augmented cost function

$$\begin{aligned} \min_{\{x_i, z_i, \zeta_i, c_i\}} & \frac{1}{2} \sum_{i=1}^N \|y_i - A_i x_i\|_2^2 + \tau_1 \|z_i\|_1 + \tau_2 \|\zeta_i\|_1 \\ & + \frac{\rho}{2} \|x_i - z_i - \zeta_i\|_2^2 + \frac{\theta}{2} \sum_{j \in \mathcal{N}_i} \|\zeta_i - c_j\|_2^2 \\ \text{s.t.} \quad & x_i = z_i + \zeta_i; \quad i \in N \\ & \zeta_i = c_j; \quad j \in \mathcal{N}_i \cup \{i\} \end{aligned}$$

- Difficult to solve by means of classical ADMM (i.e. with 2 primal iteration blocks)
- Instead, we propose to minimize the primal variables in a sequential fashion

Distributed ADMM

Algorithm

► Primal iterates

$$x_i(t+1) = (\rho I + A_i^T A_i)^{-1} (A_i^T y_i + \rho(z_i(t) + \zeta_i(t)) - \lambda_i^T)$$

$$z_i(t+1) = \mathcal{S}_{\frac{\tau_1}{\rho}} \left[(x_i(t+1) - \zeta_i(t)) + \frac{\lambda_i(t)}{\rho} \right]$$

$$\begin{aligned} \zeta_i(t+1) = \mathcal{S}_{\frac{\tau_2}{\rho + \theta |\mathcal{N}_i|}} & \left[\frac{1}{\rho + \theta |\mathcal{N}_i|} (\rho (x_i(t+1) - z_i(t+1))) \right. \\ & \left. + \theta \sum_{j \in \mathcal{N}_i} \left(c_j(t) - \frac{\mu_{i,j}(t)}{\theta} \right) + \lambda_i(t) \right] \end{aligned}$$

$$c_i(t+1) = \frac{1}{|\mathcal{N}_i|} \sum_{j: i \in \mathcal{N}_j} \left(\zeta_j(t+1) + \frac{\mu_{j,i}(t)}{\theta} \right);$$

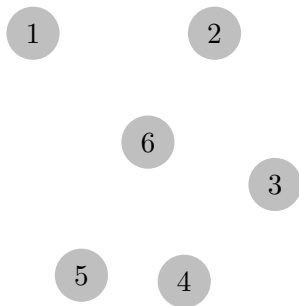
► Dual iterates

$$\lambda_i(t+1) = \lambda_i(t) + \rho (x_i(t+1) - z_i(t+1) - \zeta_i(t+1))$$

$$\mu_{i,j}(t+1) = \mu_{i,j}(t) + \theta (\zeta_i(t+1) - c_j(t+1)); \quad j \in \mathcal{N}_i$$

Distributed ADMM (cont'd)

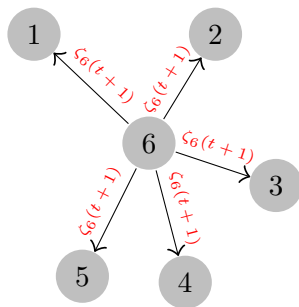
Algorithm



- Compute $\{x_i(t+1), z_i(t+1), \zeta_i(t+1)\}$ at each node

Distributed ADMM (cont'd)

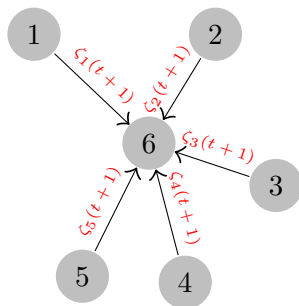
Algorithm



- Compute $\{x_i(t+1), z_i(t+1), \zeta_i(t+1)\}$ at each node
- Broadcast $\{\zeta_i(t+1)\}$ to your neighbors

Distributed ADMM (cont'd)

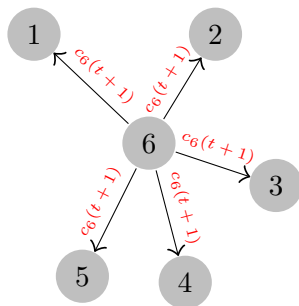
Algorithm



- ▶ Compute $\{x_i(t+1), z_i(t+1), \zeta_i(t+1)\}$ at each node
- ▶ Broadcast $\{\zeta_i(t+1)\}$ to your neighbors
- ▶ Compute $\{c_i(t+1)\}$ at each node

Distributed ADMM (cont'd)

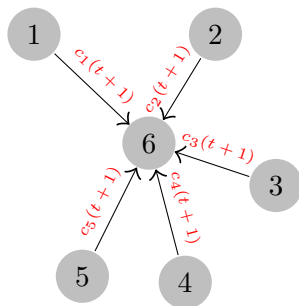
Algorithm



- ▶ Compute $\{x_i(t+1), z_i(t+1), \zeta_i(t+1)\}$ at each node
- ▶ Broadcast $\{\zeta_i(t+1)\}$ to your neighbors
- ▶ Compute $\{c_i(t+1)\}$ at each node
- ▶ Broadcast $\{c_i(t+1)\}$ to your neighbors

Distributed ADMM (cont'd)

Algorithm



- ▶ Compute $\{x_i(t+1), z_i(t+1), \zeta_i(t+1)\}$
- ▶ Broadcast $\{\zeta_i(t+1)\}$ to your neighbors
- ▶ Compute $\{c_i(t+1)\}$ at each node
- ▶ Broadcast $\{c_i(t+1)\}$ to your neighbors
- ▶ Compute $\{\lambda_i(t+1), \mu_{j,i}(t+1)\}$ at each node

Distributed ADMM with 1 bit

- ▶ DADMM requires the exchange of analog values

Distributed ADMM with 1 bit

- ▶ DADMM requires the exchange of analog values
- ▶ We propose to modify the updates of $\{\zeta_i(t+1)\}$ and $\{c_i(t+1)\}$ as follows:

$$\zeta_i(t+1) = \zeta_i(t) - \epsilon \operatorname{sign}\left(g_{\zeta_i^t}\right)$$

$$c_i(t+1) = c_i(t) - \epsilon \operatorname{sign}\left(g_{c_i^t}\right)$$

Distributed ADMM with 1 bit

- ▶ DADMM requires the exchange of analog values
- ▶ We propose to modify the updates of $\{\zeta_i(t+1)\}$ and $\{c_i(t+1)\}$ as follows:

$$\zeta_i(t+1) = \zeta_i(t) - \epsilon \operatorname{sign}\left(g_{\zeta_i^t}\right)$$

$$c_i(t+1) = c_i(t) - \epsilon \operatorname{sign}\left(g_{c_i^t}\right)$$

- ▶ $\{g_{\zeta_i^t}, g_{c_i^t}\}$ stand for the subgradients with respect to $\{c_i\}$ and $\{\zeta_i\}$ at time t

Distributed ADMM with 1 bit

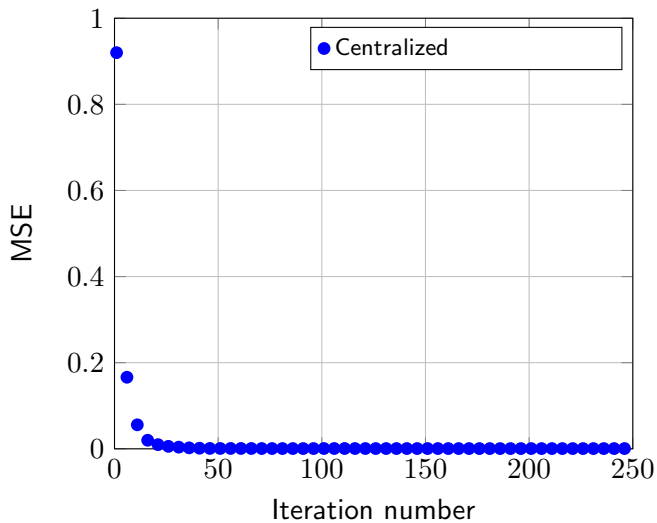
- ▶ DADMM requires the exchange of analog values
- ▶ We propose to modify the updates of $\{\zeta_i(t+1)\}$ and $\{c_i(t+1)\}$ as follows:

$$\zeta_i(t+1) = \zeta_i(t) - \epsilon \operatorname{sign}(g_{\zeta_i^t})$$

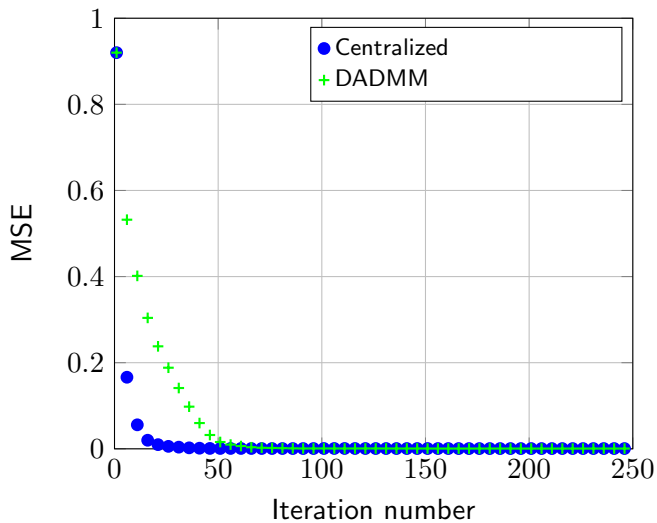
$$c_i(t+1) = c_i(t) - \epsilon \operatorname{sign}(g_{c_i^t})$$

- ▶ $\{g_{\zeta_i^t}, g_{c_i^t}\}$ stand for the subgradients with respect to $\{c_i\}$ and $\{\zeta_i\}$ at time t
- ▶ Only requires the exchange of 1-bit per variable!!!!

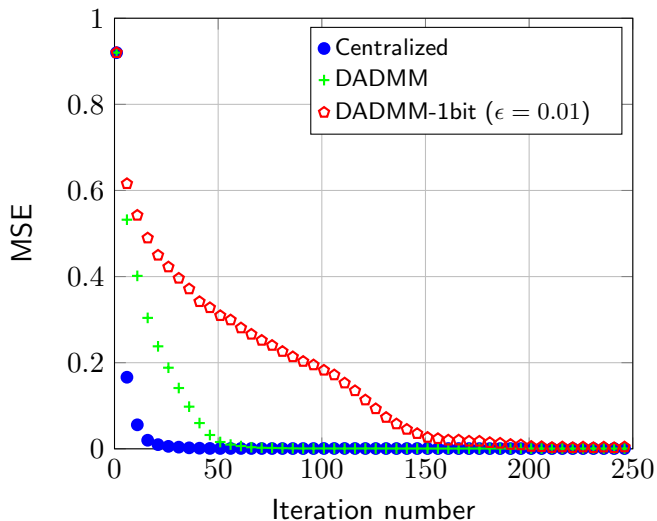
Numerical results



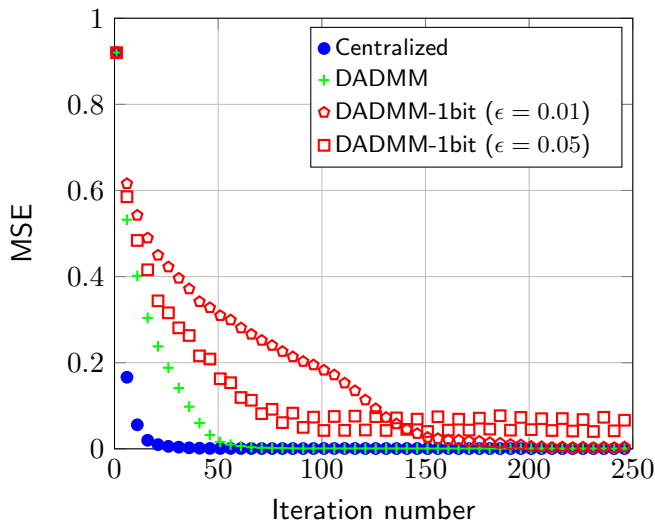
Numerical results



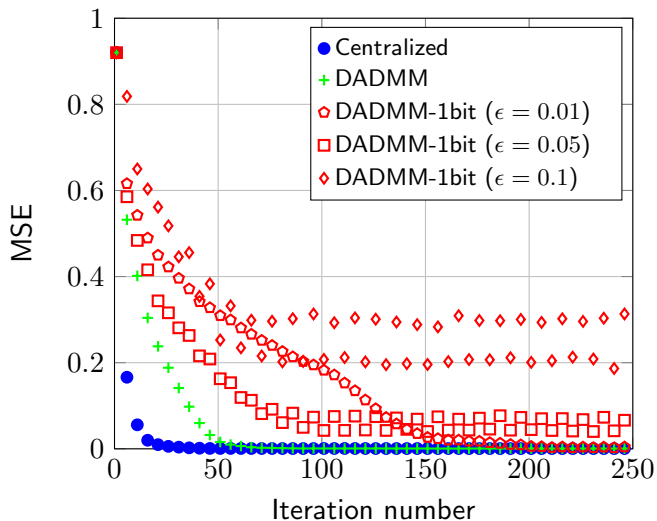
Numerical results



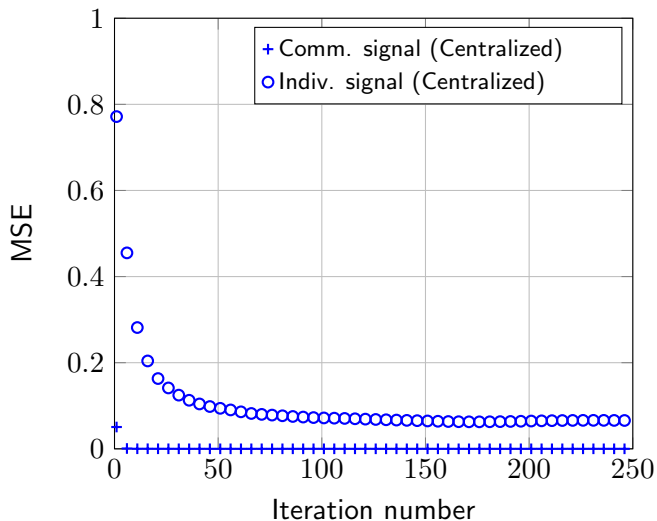
Numerical results



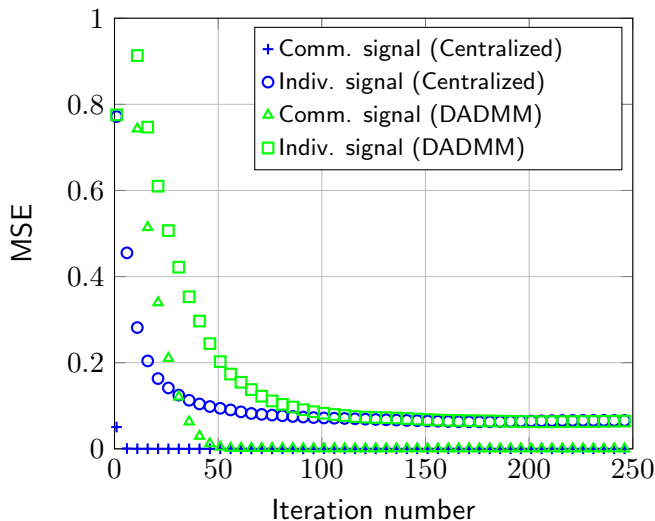
Numerical results



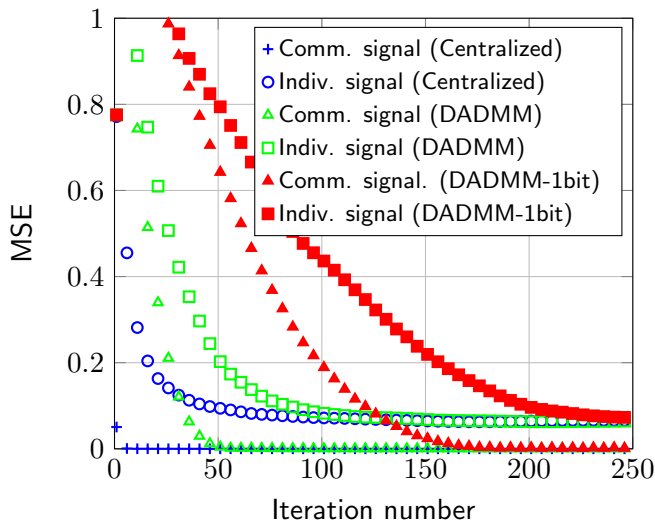
Numerical results (cont'd)



Numerical results (cont'd)



Numerical results (cont'd)



Conclusions

- ▶ We have addressed the problem of reconstruction of jointly sparse signals with innovations
- ▶ 1 Centralized ADMM solution and 2 distributed ADMM solution for in-network reconstruction have been proposed
- ▶ Distributed versions are shown to converge to the centralized ADMM
- ▶ The 1 bit version is shown to reduce the number of transmitted bits significantly

Thank you!
Questions?